MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2018/2019

ECP4136 – JAVA TECHNOLOGY

(RE / TE)

29 MAY 2019
9:00 a.m - 11:00 a.m
(2 Hours) – OPEN BOOK

INSTRUCTIONS TO STUDENTS

1. This Question paper consists of 6 pages including cover page with 3 Questions only.
2. Attempt **ALL** questions. The distribution of the marks for each question is given.
3. This is an Open-Book examination.
4. Please write all your answers in the Answer Booklet provided.

Instructions:

Figure 1 shows the UML diagram of `calculator` package, `gui` package, `BatchProcessInterface` interface, `Vector` class, `Operation` class, `BatchProcess` class, `UIFrame` class, and `ButtonHandler` class. The UML diagram will be referred in Question 1, 2 and 3.

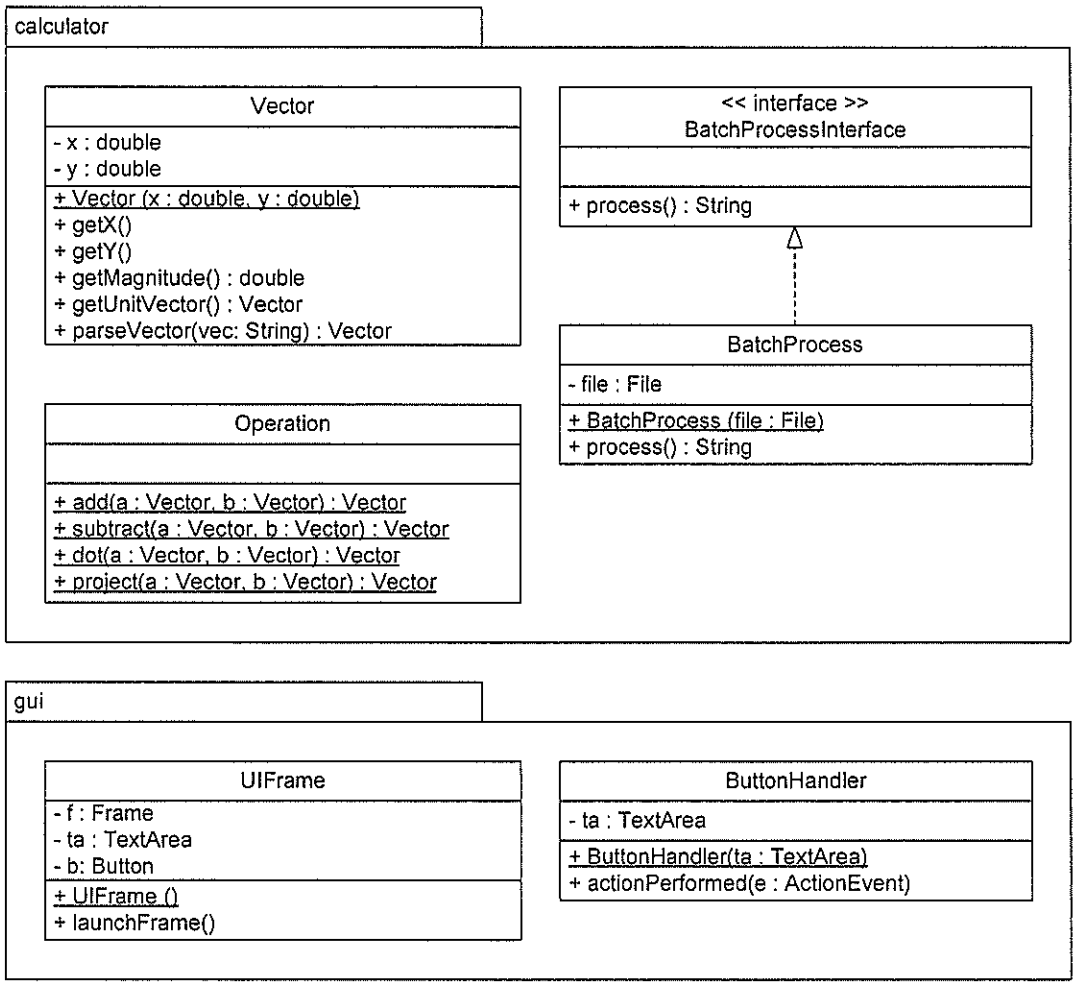

Figure 1: UML Diagram

Continued .....

Question 1

a) Implement the `Vector` class with the following requirements:

- Create the attributes specified by the UML diagram shown in Figure 1.
- Create the constructor that will save all parameters in the arguments. The parameters are the x-component and y-component of a vector.
- Implement `getX` method to retrieve the `x` attribute.
- Implement `getY` method to retrieve the `y` attribute.
- Implement `getMagnitude` method to compute the magnitude of the vector.
- Implement `getUnitVector` method to compute the unit vector. Note that a unit vector is also a vector.
- `parseVector` method parses a vector in `String` format. The following is an example of vector in `String`.

"<3,4>"

- Override `toString` method to return a `String` of the vector in the above format.

[1+2+1+1+1+2+3+1 marks]

b) The `Operation` class consists of static methods that do vector addition, subtraction, dot product and projection. Consider vector $\boldsymbol{a}$ and vector $\boldsymbol{b}$ are the two operands. Implement the `Operation` class with the following requirements.

- The `add` method adds two vectors, such that
$$\boldsymbol{a} + \boldsymbol{b}$$
- The `subtract` method performs vector subtraction, such that
$$\boldsymbol{a} - \boldsymbol{b}$$
- The `dot` method performs dot product, such that
$$\boldsymbol{a} \cdot \boldsymbol{b}$$
- The `project` method performs vector projection, such that
$$(\boldsymbol{a} \cdot \boldsymbol{b})\hat{\boldsymbol{b}}$$

[1+1+1+3 marks]

c) Create the `BatchProcessInterface` specified by the UML diagram shown in Figure 1.

[2 marks]

Continued .....

Question 2

a) Create the `BatchProcess` class to implement the `BatchProcessInterface` with the following requirements.
- Create the attribute specified by the UML diagram shown in Figure 1.
- Create the constructor that will save the `File` object in the arguments.
- The following are the requirements for the `process` method.
  - The `process` method will open and read a text file that consists of a series of commands. Every command has 3 fields separated by a space character. The first field indicates the type of operation. The second field and the third field are the two vector operands. Figure 2 shows the content in the text file.
  - The `process` method will catch `Exception`. If an exception occurs, the method will return a string "`Exception Occurs.`".
  - The process will operate the commands line-by-line. After processing all the commands, the method will return a `String` that contains the results of computations as shown in Figure 3.

[1+1+9 marks]

b) Write a console program to open a text file named "batch.txt" and use `process` method in `BatchProcess` class to execute the commands in the text file. The results are then printed on the console.

[4 marks]

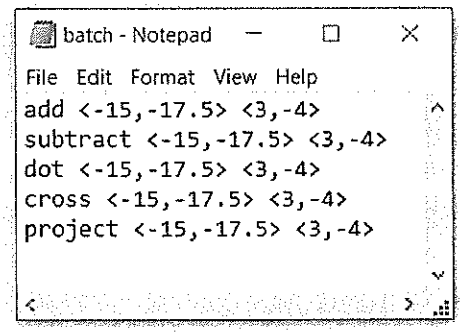

Figure 2

Continued .....

```
add <-15,-17.5> <3,-4>: <-12.0,-21.5>
subtract <-15,-17.5> <3,-4>: <-18.0,-13.5>
dot <-15,-17.5> <3,-4>: 25.0
cross <-15,-17.5> <3,-4>: Undefined Command
project <-15,-17.5> <3,-4>: <3.0,-4.0>
```

Figure 3

Question 3

a) Create the `UIFrame` class specified by the UML diagram shown in Figure 1. `UIFrame` can be launched to present the user interface as shown Figure 4. The following are the requirements.

- The name of the `Frame f` is "Vector Computations".
- The `Frame f` uses `BorderLayout`
- The `TextArea ta` consists of 10 rows and 50 columns.
- `WindowListener` is added to the `Frame f` to terminate the program when the close button is clicked.
- The button is named as "Process".
- When the Process button is clicked, the action is handled by `ButtonHandler`.
- The components are added to the `Frame f` in the correct way.

[1+1+1+3+1+1+2 marks]

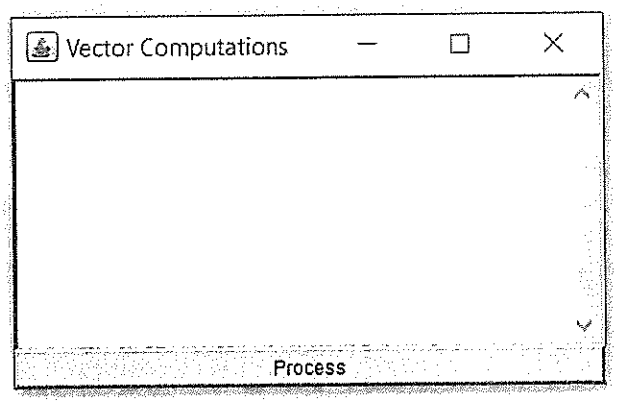

Figure 4

Continued .....

b) Create the `ButtonHandler` class with the following requirements.
- Create the attribute specified by the UML diagram shown in Figure 1.
- Create the constructor that will save the `TextArea` object in the arguments.
- Implement the `actionPerformed` method with the following requirements.
  - The method opens a text file named "batch.txt" and use `process` method in `BatchProcess` class to execute the commands in the text file.
  - The results are then printed on the `TextArea` object.

[2+1+2 marks]

End of Page